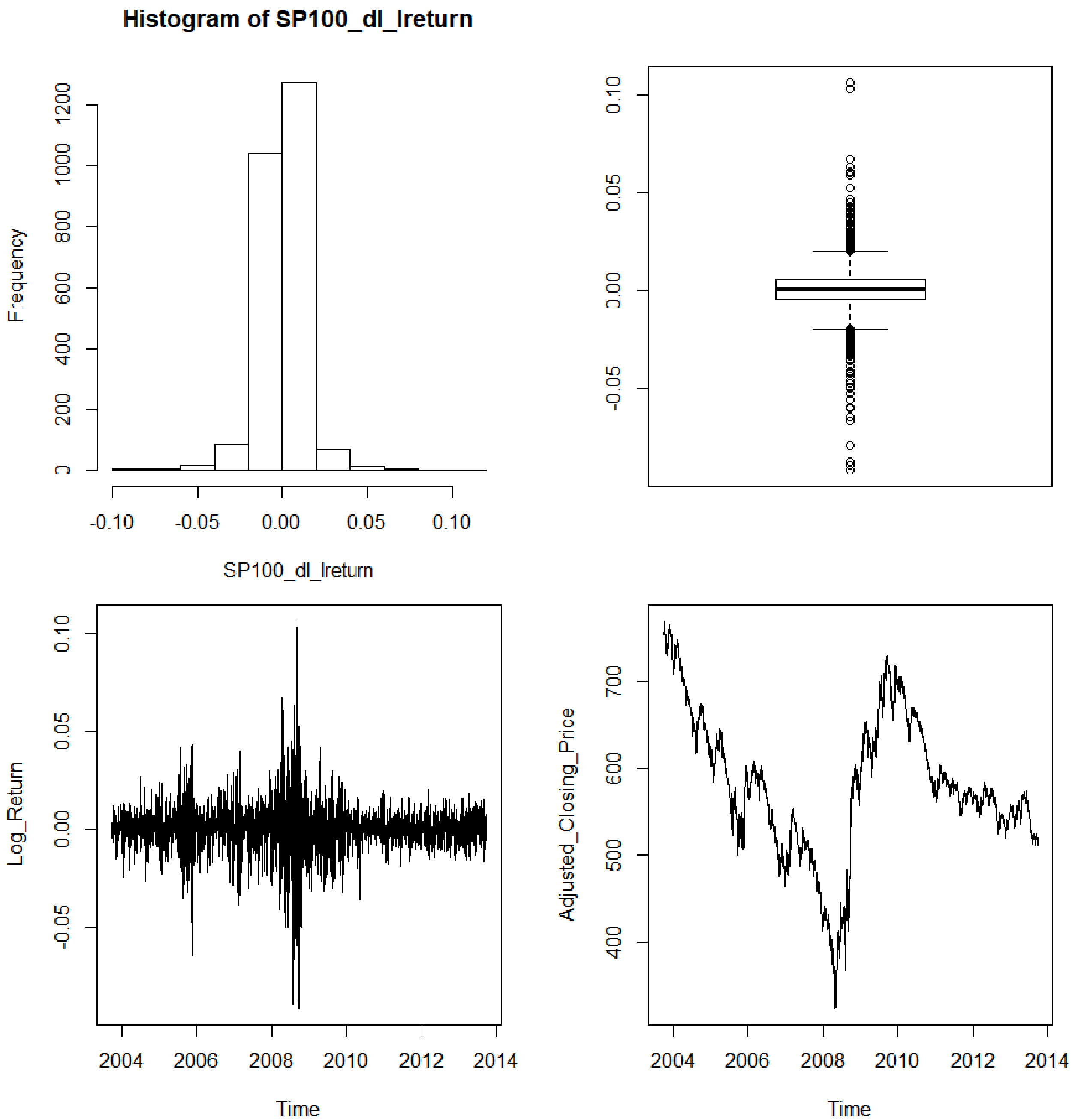
STATS 509 Homework 2 Xiaofeng Nie

1. Solution:

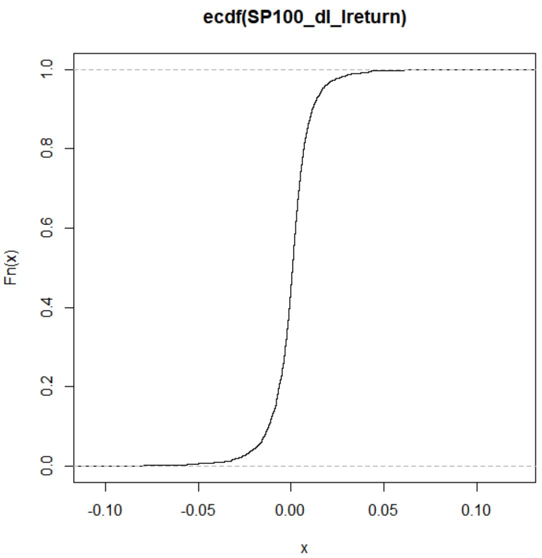
a) Descriptive statistics shows that the mean and of S&P index log return is almost 0, and there is slightly left skewness. Considering the histogram, boxplot and kurtosis value—11.38, we can say that the distribution of log returns is leptokurtic and heavy tailed.

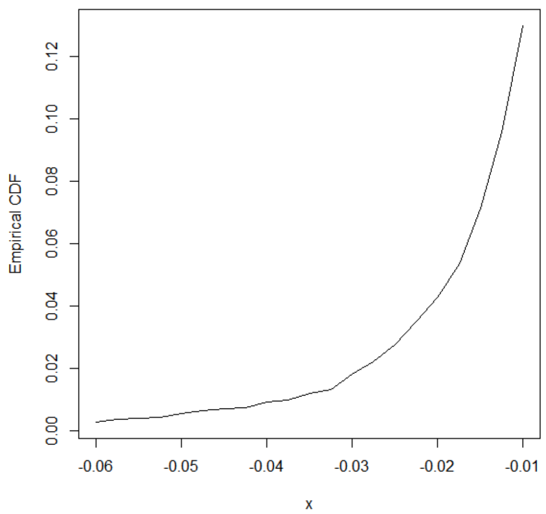
From the plot of log returns and adjust price of S&P index, we can find that in 2008 there is a severe drop leading to the extreme value of log returns.

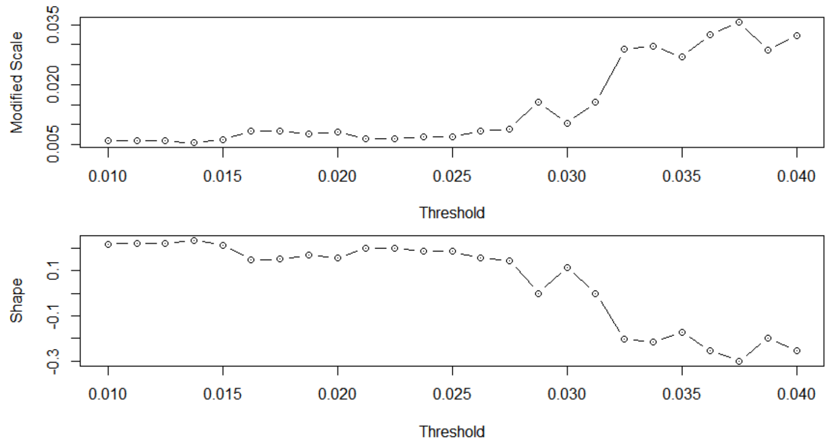
Statistics (of log-return)	Mean	Median	Standard Deviation	Skewness	Kurtosis
Value	0.0001549	0.0007256	0.01242858	-0.2717636	11.37748

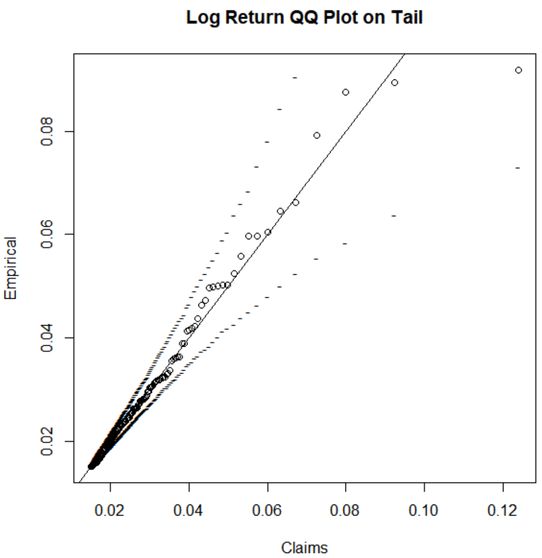


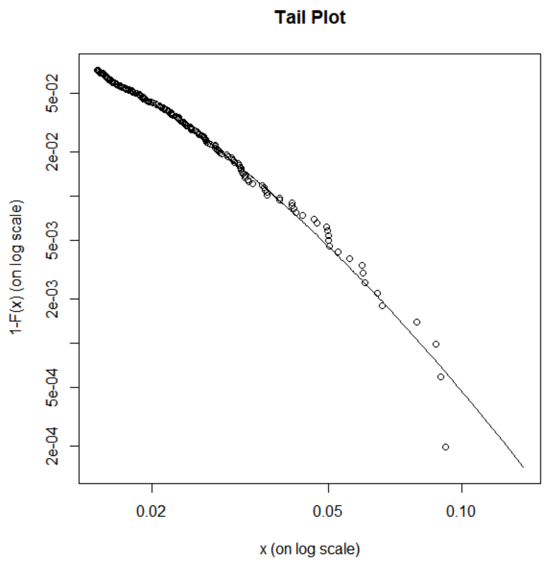
b) From the shape plot of threshold, we can find that, in the interval (0.010,0.025), the plot is stable. Thus, we can choose 0.015 to be the threshold. Q-Q plot and tail plot show that all the points are near lines, which means the GPD model of 0.015 threshold is quite good.







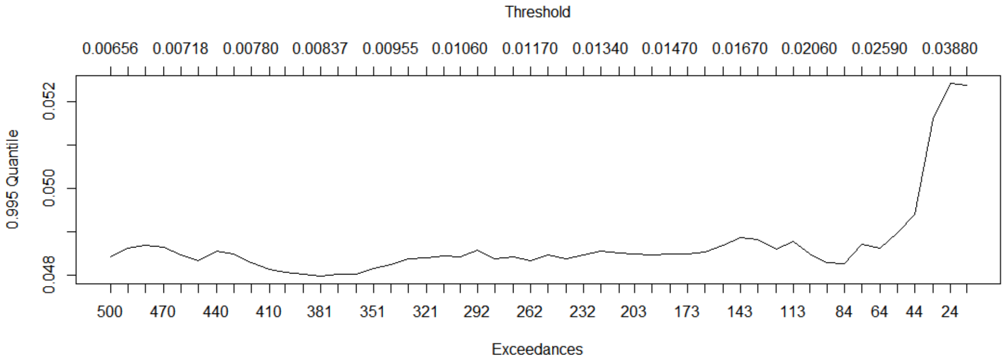




c) When log return is normal distributed, $r_{0.005} \approx -0.03185897 \Rightarrow R_{0.005} = \exp(-r_{0.005}) - 1 \approx -0.03135682$, i.e. the relative VaR is 0.03135682. This may not be precise, because from (a) we know that the distribution of log return is leptokurtic and heavy tailed, rather than normal distributed.

d) When using GPD distribution, $r_{0.005} \approx 0.0484558 \Rightarrow R_{0.005} = 1-\exp(-r_{0.005}) \approx 0.04730055$, i.e. the relative VaR is 0.04730055, which is larger than relative VaR computed from normal distribution.

e) Relative VaR is stable in the threshold interval (0.00656,0.02), when threshold is larger than 0.02, relative VaR will change severely.



f) Using Monte-Carlo simulation, expected shortfall will be 0.0682569*(value of portfolio).

2. Solution:

(a) $E[0.6R_1 + 0.4R_2] = 0.6ER_1 + 0.4ER_2 = 0.6 * 0.02 + 0.4 * 0.03 = 0.024$

$$\begin{aligned} Var[0.6R_1 + 0.4R_2] &= 0.6^2 Var(R_1) + 0.4^2 Var(R_2) + 2 * 0.4 * 0.6 * \rho * \sqrt{Var(R_1)Var(R_2)} \\ &= 0.6^2 * 0.03^2 + 0.4^2 * 0.04^2 + 2 * 0.4 * 0.6 * 0.5 * 0.03 * 0.04 \\ &= 0.000868 \end{aligned}$$

(b)

$$\begin{aligned} Var(wR_1 + (1-w)R_2) &= w^2 Var(R_1) + (1-w)^2 Var(R_2) + 2w(1-w)\rho\sqrt{Var(R_1)Var(R_2)} \\ &= (1.3w^2 - 2w + 1.6) * 10^{-3} \end{aligned}$$

$$\frac{dVar}{dw} = (2.6w - 2) * 10^{-3} = 0 \Longrightarrow w = \frac{10}{13} \approx 0.7692$$

Thus, w=0.7692 can minimize variation of the linear combination.
Because variation means risk, when we choose the portfolio with minimized variation, we also choose the portfolio with minimized risk.

(c) Expected shortfall is the maximum expected loss of the portfolio. The portfolio with value w that minimizes the expected shortfall have the minimized maximum expected loss.
When w=0.69, the VaR is 51507.06, and the minimized expected shortfall is 60657.95.

(d) When w=0.73, the VaR is 62609.54, and the minimized expected shortfall is 197647.3.

Appendix:

R code for problem 1

```
library(fBasics)
library(POT)
library(evir)
#a)
dat = read.csv("O:\\18WIN\\STATS 509\\HW3\\SP100_daily_03-13.csv",header
= T)
SP100_dl = rev(dat$AdjClose)
SP100_dl_lreturn = diff(log(SP100_dl)) # generating log returns (daily)
d = as.Date(dat$Date,format="%m/%d/%Y")
summary(SP100_dl_lreturn)
sd(SP100_dl_lreturn)
skewness(SP100_dl_lreturn)
kurtosis(SP100_dl_lreturn)
par(mfrow=c(1,2))
hist(SP100_dl_lreturn)
boxplot(SP100_dl_lreturn)
plot(d[-1],SP100_dl_lreturn,type="l",xlab="Time",ylab="Log_Return")
plot(d,SP100_dl,type="l",xlab="Time",ylab="Adjusted_Closing_Price")
#b)
par(mfrow=c(1,2))
eecdf = ecdf(SP100_dl_lreturn)
plot(eecdf)
uv = seq(from=-0.06, to=-0.01, by=0.0025)
plot(uv,eecdf(uv),type="l",xlab="x",ylab="Empirical CDF")
tcplot(-SP100_dl_lreturn,c(0.01,0.04),nt=25,conf=0)
gpd_fit=fitgpd(-SP100_dl_lreturn,0.015)
qq(gpd_fit, main="Log Return QQ Plot on
Tail",xlab="Claims",ylab="Empirical",ci=TRUE)
gpd_est=gpd(-
SP100_dl_lreturn,thresh=0.015,method=c("ml"),information=c("observed"))
tp=tailplot(gpd_est,main="Tail Plot")
#c)
r_0.005 = qnorm(0.005,mean(SP100_dl_lreturn),sd(SP100_dl_lreturn))
R_0.005 = exp(r_0.005)-1
r_0.005;R_0.005
#d)
m = 0.015
alphat = 1-0.005/eecdf(-m)
xi = gpd_est$par.ests[1]
scale = gpd_est$par.ests[2]
r = qgpd(alphat, xi, m, scale)
R = 1-exp(-r)
r;R
#e)
quant(-SP100_dl_lreturn, p=0.995, models=50,reverse=TRUE, ci=FALSE,
auto.scale=TRUE, labels=TRUE)
#f)
niter = 1e4
exp_shortfall = rep(0,niter)
set.seed(2015)
for (m in 1:niter)
{
  r = rgpd(1000,xi,0.015,scale)
  index = which(r > R)
  exp_shortfall[m] = mean(r[index])
}
mean(exp_shortfall)
```

R code for problem 2

```
library(VaRES)

#c
w = seq(0, 1, by=0.01)
mu = 0.02*w+0.03*(1-w)
var = 0.03^2*w^2+0.04^2*(1-w)^2+2*0.03*0.04*0.5*w*(1-w)
q = rep(0,length(w))
for (i in 1:length(w)) {
  q[i] = esnormal(0.005,mu[i],sqrt(var[i]))
}
num = which.max(q)
"weight";w[num]
"expected_shortfall";(-q[num])*1e6
"VaR";-qnorm(0.005, mu[num], sqrt(var[num]))*1e6

#d
shortfall = rep(0,1000)
exp_shortfall = rep(0,length(w))
set.seed(2015)
for (i in 1:length(w)) {
  lambda = sqrt(var[i])*sqrt(4/6)
  t_std = qt(0.005, 6)
  t_new = t_std*lambda+mu[i]
  q[i] = exp(t_new)-1
  rt = rt(1000, 6)
  rt_new = rt*lambda+mu[i]
  for (n in 1:1000){
    return = exp(rt_new)-1
    index = which(return < q[i])
    shortfall[n] = mean(return[index])
  }
  exp_shortfall[i] = mean(na.omit(shortfall))
}
num = which.min(exp_shortfall)
"Weight";w[num]
"expected shorfall";-exp_shortfall[num]*1e6
"VaR";-q[num]*1e6
```